THE DESIGN AND IMPLEMENTATION OF THE REGISTRAR'S INFORMATION SYSTEM AT THE NAVAL POSTGRADUATE SCHOOL

by

Charles David Vincent

United States
Naval Postgraduate School

THESIS

The Design and Implementation of the Registrar's Information System at The Naval Postgraduate School

by

Charles David Vincent

Thesis Advisor: E. A. Singer

March 1971

Approved for public release; distribution unlimited.

The Design and Implementation of the Registrar's Information System at The Naval Postgraduate School

by

Charles David Vincent
Lieutenant Commander, United States Navy
A.B., Harvard College, 1957

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
March 1971

ABSTRACT

The design and implementation of a new information system for the Registrar's office at the Naval Postgraduate School was preceded by an analysis of the original punched-card record-keeping operation. Major goals for the new system included rapid grade reporting, feedback reports to increase data reliability, file security and disaster recoverability, data retrieval from current and historical data bases, and ease of input-output batch processing. Implementation was accomplished on an IBM System/360 Model 67 operating under OS/360. The information system utilized direct access storage devices and list-processing methods. Additional reports for professors, administrators, and students are planned utilizing the course, registration, professor, student, entrance-credit, and thesis-title files maintained on direct-access disk units. File expansion and future interfaces for the Registrar's information system are discussed.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

The author is deeply indebted to many at the Naval Postgraduate School for assistance during the months of preparation for this project.

A large debt of gratitude is owed the Registrar of the Naval Postgraduate School, Miss Jean Warriner. Her patience in unraveling the mysteries of the original punched-card system and her endurance during the design of the new system are deeply appreciated. The extra work caused by the conversion to the new and the parallel run with the old system has taxed the whole staff of the Registrar's Office.

Thanks are also due to Dean B. J. Lockhart for his interest and support of the program and his helpful suggestions during critical phases of the work.

The computer center provided unusual support in the person of Mr. Clarence Kellogg, who spent many hours and weekend nights helping the author in the debugging and operation of the computer programs. It will be through Mr. Kellogg's continuing effort that the new system can be completed.

The author is indebted to his advisor, LCDR Ed Singer for his guidance and his help in resolving some of the difficult problems of the new system.

Finally, the author alone is responsible for any errors in this report.

I. INTRODUCTION

The record-keeping system of the Registrar's Office at the Naval Postgraduate School was developed piecemeal over the past 10 years as a punched-card file operation. As the student enrollment of the school increased, the punched-card files became large, unwieldy, and not conveniently accessed except by hand search. The installation of second and third-generation computers at the School only served the Registrar's data-processing needs by speeding up the printing process.

This study attempted an overall analysis of the Registrar's academic record-keeping operation from a systems point of view. The use of random access capability of disk storage was incorporated in the file organization in order to improve the present and future data retrieval needs of the school.

The design and implementation of the Registrar's information system proceeded using the following plan:

1. Analyze present system.
2. Establish objectives for new system.
3. Set output formats.
4. Establish input processing routines and command language format.
5. Program and debug new system.

6. Convert present punched-card files and implement updating procedures.
7. Test new system in parallel with present system.
8. Release new system for operation.

At the time of this report Steps 1 through 4 have been completed and are reported herein. Steps 5, 6, and 7 have all been started and are progressing. Step 6 is practically completed except for a few minor items. The feasibility of the overall system has been demonstrated by extensive computer programming. The remaining programming should be completed within three to six months by Computer Center staff personnel. Step 8 should not be attempted until the parallel run has proven the reliability and versatility of the new system.

II. ANALYSIS OF THE ORIGINAL ACADEMIC RECORD SYSTEM

The first step in the systems analysis of the record-keeping function of the Registrar's Office was the analysis of the original punched-card system. This step was further subdivided into the following investigations:

1. Organizational entities which have a bearing on the information flow of the present system.
2. The processing cycle through one academic quarter.
3. Inputs and outputs.
4. Card file organization and processing.
5. Discussions with users and providers of data (curricular officers, deans, department chairment, students, and professors).
6. Summary of problems with present system.

A. ORGANIZATIONAL ENTITIES

The organizational relation of the Registrar's Office in the academic organization of the Naval Postgraduate School is shown in Figure 1. The Registrar's Office is staffed with three full-time workers, including the Registrar, and one part-time worker. Programming assistance is provided by the Computer Center. Keypunching of input data is accomplished within the Registrar's Office and occupies the full-time efforts of one worker and part-time efforts of the other workers. The other organizational entities which have a direct relationship to the information flow of

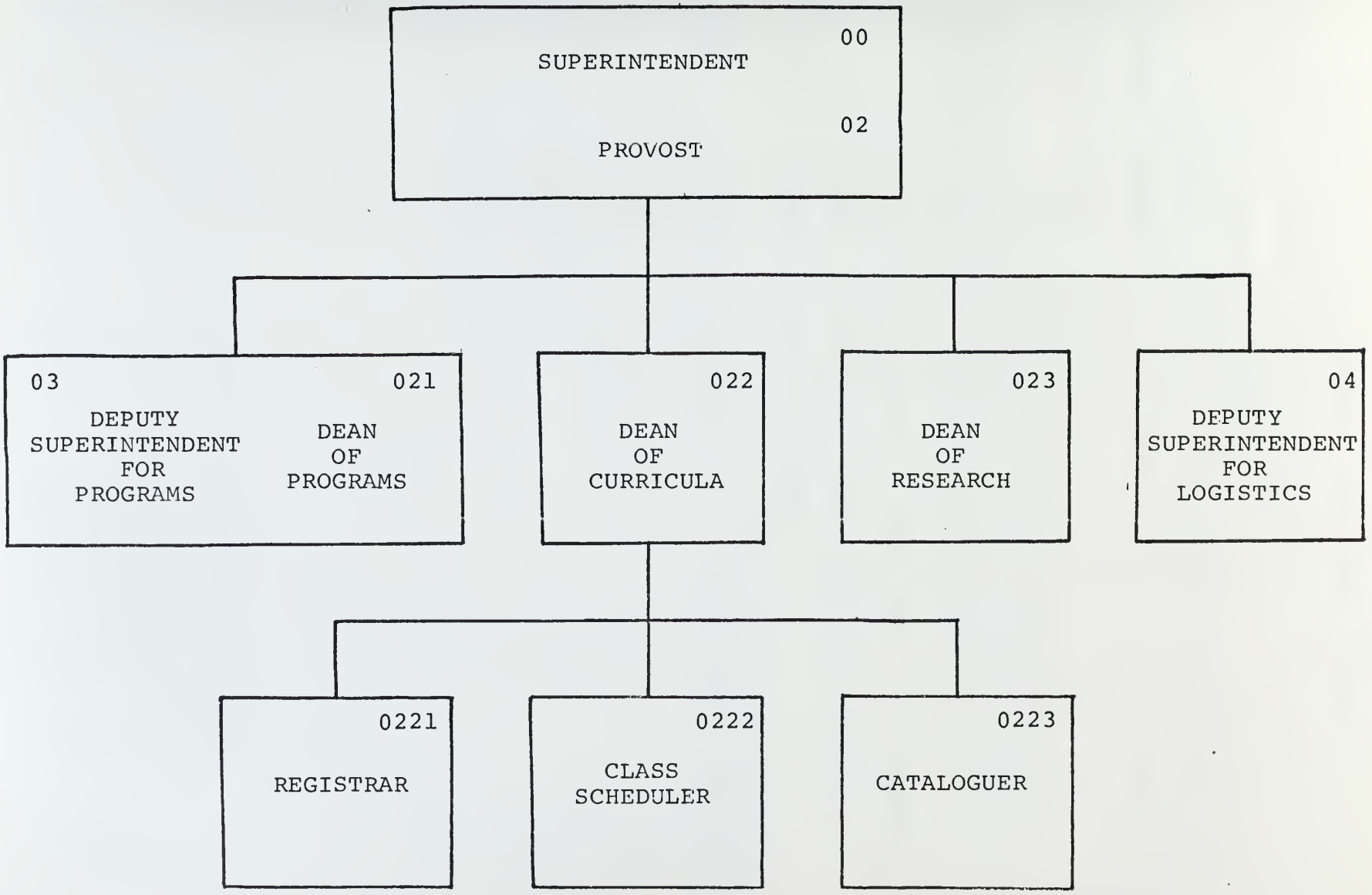

Figure 1. Academic Organization of the Naval Postgraduate School.

the Registrar's Office are the curricular officers, the academic departments, and the students.

1. The Curricular Officers

Directly reporting to the Deputy Superintendent for Programs are the nine curricular officers who are responsible for the following functions:

> "(1) academic and military supervision and direction of officer students; (2) coordinating, in conjunction with Academic Associates, the elements of each curriculum within their program areas; and (3) conducting liaison with curricula sponsor representatives."[1]

The curricular officers, along with their organization code, number of curricula supervised, and number of students supervised, are as follows:

Code	Curricular Officer	# Curricula Supervised	# Students Supervised[2]
30	Operations Analysis	2	380
31	Aeronautical Engineering	1	119
32	Electronics and Communications Engineering	3	280
33	Ordnance Engineering	4	190
34	Naval Engineering	1	100
35	Environmental Sciences	2	226
36	Management & Computer Science	3	431
37	Engineering Science	1	174
38	Baccalaureate	2	343
	Totals	18	2,243

---

[1] Naval Postgraduate School Catalogue for 1970-1972, p. 9.

[2] The student figures were derived from the estimated total number of students to be enrolled during Fiscal Year 1970. The source was U.S. Naval Postgraduate School, Integrated Operating and Development Plan, 1970, pp. III-1 and III-2. The figure does not include an additional 103 Immediate Graduate Education Students (IGEP's), who were also supervised by the various curricular officers.

2. The Academic Departments

The faculty of the Naval Postgraduate School is organized into eleven academic departments, each supervised by a civilian department chairman. The departments, along with their administrative code and number of professors, are as follows:

Code	Department	No. of Professors[3]
51	Meteorology	16
52	Electrical Engineering	48
53	Mathematics	39
54	Material Science and Chemistry	13
55	Operations Analysis	42
56	Government and Humanities	12
57	Aeronautics	22
58	Oceanography	16
59	Mechanical Engineering	15
61	Physics	30
62	Business Administration and Economics	33
	Total	286

3. The Officer Students

The large majority of students at the Naval Postgraduate School are student officers ordered to the school by the Navy, Marine Corps, Coast Guard, Army, Air Force, and twenty-three allied nations. In addition, a few members of the civilian and military staff attend and obtain academic credit for courses. The Aviation Safety Program, also conducted by the Naval Postgraduate School, normally convenes

---

[3]The number of professors was derived from a list of all professors who have taught courses during the first three quarters of fiscal year 1970-1971. This list was used to set up the master professor file discussed in Section VI of the thesis.

four classes per year of about twenty-five students each. The Registrar maintains academic records for them as well.

The military officers attending classes are organized into military sections by their assigned curricular officers, and most administrative business is conducted via the senior member of each student section. Each student is also assigned a Student Mail Center (SMC) box and is expected to pick up his mail once each working day.

4. Organizational Changes

The organizational entities described above represent a structure at only one point in time. This structure is undergoing gradual change, and any data processing system should be designed to cope with these changes. For example, the number of academic departments increased from ten to eleven over the past two years. The merger of two academic departments is planned for 1 July 1971. The estimated average on board student loading is planned to increase by fifty-six during Fiscal Years 1973-1978,[4] and the number of professors is planned to be increased by forty-seven during this same period.[5]

B. ACADEMIC QUARTER PROCESSING CYCLE

The fifty-two weeks of one academic year are divided into four academic quarters of twelve weeks each, plus two

---

[4] Ibid., p. III-3

[5] Ibid., p. IV-2.

inter-quarter breaks of two weeks each. These breaks are scheduled for July and December of each year. Final examinations are scheduled during the twelfth week of each quarter. The academic year begins in July, and the successive quarters are numbered 1, 2, 3, and 4. For computer processing purposes the quarters are prefixed by a two-digit year. For example, Quarter 701 was the first quarter of academic year 1970-1971, Quarter 703 is the current academic quarter, which convened in January, 1971, and ends in March, 1971. Because there was no inter-quarter break in September, 1970, Quarters 701 and 702 are designated "back-to-back" quarters. Similarly, Quarters 703 and 704 will be "back-to-back."

An officer student normally attends classes in successive quarters from the time he commences his curriculum study until he completes his degree requirements or is awarded a certificate of course completion. The time of study varies, depending on the curricula, from four to eighteen quarters. The median student is on board for six quarters. New students arrive each quarter, and there is a graduation and awarding of degrees once each quarter.

The basic processing cycle of the Registrar is one quarter but proceeds over a period of more than one calendar quarter. At any one point in time there is processing underway for more than one quarter, the actual schedule depending on whether the quarter is a back-to-back or not.

The following is a description of overall operations of the original punched-card system.

1. Week 9: Roster Letter

A listing of all students in the master card file was sent to the curricular officers for updating during the ninth week. Under the original system this was the main updating device used to correct student information such as section assigned and change of rank. The primary purpose of the list, however, was to verify the students who were expected to graduate at the end of the next academic quarter. These potential graduates were identified by the curricular officer with a check mark opposite the student name. After the roster letters were returned to the Registrar, the student master file was updated with the changes, and the cards for the potential graduates pulled by hand and placed in a separate group.

2. Week 10: Preliminary Input List

Based on copies of student orders to the school and also on information returned by the curricular officers on the roster letter, a list of new students was published. Thirteen copies of this list were distributed.

3. Week 12: New Quarter Course and Header Cards

During the last week of each quarter six registration cards were prepared for each student expected to be on board for the next quarter. These cards contained quarter number, section number, officer file number, designator, rank, corps/country code, alpha sequence number, and student

name. Potential graduate cards were produced with a distinguishing color. These cards were distributed via the curricular officers and section leaders to the students on or before the first day of classes. They were turned in by the students to the professors of each course in which they should be registered. The actual schedule for each student was determined by the curricular officer in conjunction with the Master Schedule of Classes published by the Class Scheduler. Under the original system the student actually registered himself by handing in one of these course cards.

Also during the twelfth week, or as soon as the Master Schedule of Classes was published, header cards for each course segment scheduled were prepared and distributed to the academic departments. These header cards were combined by the secretaries of the various departments with the student course cards turned in by the instructors and all were returned to the Registrar during the first, second, and third weeks of the new quarter.

4. Week 2: Temporary Class Rosters

From the class header cards and student (detail) cards turned in by the academic departments, preliminary class rosters were prepared. The course information (course name, course number, segment number, lecture hours, and lab hours) from the header card was duplicated onto the detail (student) cards. The groups of header and detail cards were manually filed in a group by course number. These cards new made up the current quarter's registration file. A complete file for one quarter constituted about 7000 cards.

5. Week 3: Permanent Class Rosters

Two copies of the Temporary Class Roster were produced from the original course cards and sent to the instructor. One copy was retained for the instructor's use in administering the class. The second copy was verified by the instructor and returned to the Registrar with any corrections noted. The punched cards were updated with the corrections, and 3 copies of the Permanent Class Roster produced, 2 for the instructor and one for the Dean of Programs.

6. Week 5: Potential Graduates List and On Board List

Eighteen copies of the list of potential graduates were produced and distributed. Thirty copies of the on board list were produced and distributed. Each list contained file number, designator, rank, corps/country code, name, and education code (four digits). The lists were titled with the as-of date and page number. A total number of students on the list appeared at the end.

7. Week 9: Incomplete Grade Reminders

A memorandum was sent to all studnets who had incomplete grades outstanding from the previous quarter, with a copy to each curricular officer and professor. All incomplete grades had to be changed to a letter grade by the end of the twelfth week, or else they were administratively changed to failure.

8. Week 11: Grade Rosters

Another copy of the class roster was produced from the registration file in order for the instructors to report

the course grades. This copy was distributed via the academic departments and was required to be returned no later than 1000 on Monday following the end of the quarter (twelfth week). Although early submission of grades was encouraged, the great bulk of the grade rosters were actually turned in between 0800 and 1000 of the last possible day.

9. Week 1: Grades and Transcripts

The first week following the end of the quarter was the peak processing time for the Registrar. While the grade rosters were being received, the course cards were manually pulled from the registration file and placed in the keypunch. An alphabetic grade was keypunched on each student card from the grade roster. As course segments were finished, they were placed in a processing tray, taken to the computer center, and an official roster (with printed grade) was produced for verification by the instructor.

During Week 10 or 11, the student cards for the potential graduates were manually pulled and placed in a "quick processing" bin. The potential graduate cards in most cases were identified by a different color stripe. After all the courses were keypunched with grades and the official rosters run, then these cards were alphabetically sorted and assembled with the grade cards from previous quarters. It was only at this point that grade reports for graduates in the form of an academic record could be produced. These academic records with the final quality point averages for the graduates were an essential item that must

have been completed before the Academic Council could convene for the recommendation of degree awards. Until the second quarter of Academic Year 1970-1971, the Academic Council met at 1300 on Tuesday following the end of a quarter. Graduation was held the following Wednesday morning. Commencing with the second quarter of Academic Year 1970-1971, the graduation ceremony was held on Friday of the last day of the quarter. The actual awarding of degrees was accomplished after the Academic Council met later in the first week following the end of the quarter. While late-night work to produce the academic records was no longer required, there was still considerable pressure to meet the Academic Council deadline and to get the grades published for all students, as well.

Academic records were produced on three-part colored paper. The white original was filed alphabetically by student name in the Registrar's files and was used to produce all official copies. The second (yellow) copy and the third (pink) copy were both forwarded to the curricular officer. The yellow copy was retained by the curricular officer for the student's file, and the pink copy was forwarded via the section leaders to the students.

The official class rosters (with printed grades) were forwarded to the academic departments for verification and were filed in the Course Journals maintained by each department.

For each graduate a certified copy of the academic record showing degree awarded was prepared. The thesis title (if applicable), honors and other academic awards were typewritten on the original copy.

C. INPUTS TO THE ORIGINAL SYSTEM

The following is a list of the major inputs to the original system along with the source and type of card produced (or modified) and card-frequency per quarter:

	Input Form Name	Source	Card Type	No. of Cards
1.	Master Instruction Schedule	Class Scheduler	"A"	400
2.	Grade Distribution Card	Class Roster Program	"B"	400
3.	Degrees Granted	Academic Council Minutes	"E"	250
4.	Dean's List Card	Academic Record Program	"H"	252
5.	Orders for New Inputs	Naval Messages & Official Letters	"J"	250
6.	Registrar's Information Sheet (upon reporting on board)	Student	"J"	250
7.	Roster Letter	Curricular Officer	"J"	500
8.	Transcripts of Previous Academic Work	Other Universities (upon request by student)	"2"	300
9.	Credits Brought Forward on Change of Academic Year	Academic Record Computer Program	"4"	1800 (Qtr 1 only)
10.	Credits Brought Forward on Change of Curriculum	Curricular Officer via Dean of Curricula	"4"	30

11.	Course Registration Card	Student via Instructor & Academic Dept.	"5"	7000
12.	Request for Change of Registration	Curricular Officer via Instructor & Dean of Curricula	"5"	80
13.	Request for Change of Grades	Instructor via Chairman of Dept & Dean of Curricula	"5"	50
14.	Report of Intercurricular Transfer	Old Curr. Officer to Dept. Supt. via New Curr. Officer	"J"	100
15.	Application for Admission (Courses for Credit)	Staff member via Head of Dept. and Academic Dean	"J" & "5"	30

D. OUTPUTS FROM THE ORIGINAL SYSTEM

The following is a list of the major outputs from the present system along with the card file source, frequency, and number of reports per quarter:

	Output Name	Source	Frequency	Number produced (#Copies)	No.of Pages
1.	Roster Letter	"J" Cards	Q	9	12
2.	Course Cards ("5" cards)	"J" Cards	Q	1800	6
3.	Course Control Sheet	"A" Cards	Q	1	4
4.	Class Rosters	Registration File ("A" & "5" Cards)	Q	400(5)	1.5
5.	Input List	"J" Cards	Q	2(13)	7
6.	On Board List	"J" Cards	Q	1(30)	45
7.	Potential Graduates List	"J" Cards	Q	2(18)	5
8.	Academic Records (for Graduates)	Academic Record File (Cards)	Q	250(3)	1

9.	Academic Records (All Others)	Academic Record File (Cards)	Q	1600(3)	1
10.	Academic Records (Due to Grade Changes)	Academic Record File (Cards)	Q	30(3)	1
11.	Certified Transcripts	Last Academic Record	Q	600	3
12.	Dean's Lists	"H" Cards	Q	2(15)	9
13.	Dean's List Statistical Summary	"H" Cards	Q	2	1
14.	Grade Distribution Study	"B" Cards	A (Aug.)	2	25
15.	BuPers Enrollment Report	Input List	Q	1(2)	15
16.	BuPers Graduation Report	Graduate List	Q	1(27)	10

E. CARD FILE ORGANIZATION

The Registrar's card files were maintained in metal punched-card tray files in the Registrar's Office in the East Wing of Herrmann Hall. The building itself is constructed of wood and is not fireproof. For on-board students the major files were the current quarter course cards and the on-board student master file. There were other files maintained for previous academic years and for students not on board.

1. The Current Quarter Course Card File

This card file consisted of Course Header Cards ("A" Cards) and student detail cards ("5" Cards). There was one header card for each academic course in which students were registered for credit. The file consisted of about 400 header

cards and 6600 detail cards. The file was ordered by course number which consisted of two alpha characters followed by four digits and a one-digit segment number.

After the grades were punched on the "5" cards at the end of the quarter and the final class rosters produced, the dismantling process began. The file was sorted on the three-digit sequence number and the first character of the last name to bring all "5" cards for one student together. These cards were then manually inserted at the end of the academic record group for each student, and the academic records produced, first for graduates and then for the rest of the students. This process took two to three days. The resulting academic records which were printed on three-ply paper were then burst and sorted by curricular officers for final delivery. The original copy was retained in alphabetical order and filed in the current student files in the Registrar's Office. The graduates' copies were held for microfilming at the end of the Academic Year. The microfilm was stored in another building for safe-keeping.

2. The On-board Student Master File

This file constituted the academic record for all on-board students during any one academic year. The file was started anew each July with master "J" cards for each on-board student. Balance-forward cards ("4" cards) were produced by the previous running of the academic records and produced the continuity necessary to update the quality

point averages for these students. As the year progressed, the following additional cards were added to the file:

Card Type Added	Purpose
"2"	Lists entrance credits from other schools
"5"	Adds completed courses and grades to record
"J"	Starts a new curriculum for curriculum changes
"E"	Lists the degree and date the degree was granted.

The size of the file increased from 4,000 cards at the beginning of the fiscal year to about 35,000 cards at the end of the fiscal year.

3. Control Files

A set of master "J" cards was maintained for all students on board. This file was subdivided into the following segments:

a. Potential Inputs Not Yet On-board (by quarter of input)

b. Inputs That Have Arrived

c. On-board Students

d. Potential Graduates

4. Historical Files

As a current academic year ended and as students departed for various reasons, their academic record files were placed in "dead files" as follows:

a. Academic Record Cards for On-board Student (past academic years)

b. Academic Record Cards for Students Departed (Grouped by Quarter of Departure)

c. Course Header Cards for Course Segments Given in the Past

d. "J" Control Cards for Students No Longer on Board

F. DATA USER INTERVIEWS

In order to obtain an understanding of how the data produced by the Registrar's Office was used and to solicit views on changes to the original system, a series of interviews and briefings was conducted as follows:

1. A general briefing was held for all curricular officers, the Dean of Curricula, and the Deputy Superintendent for Operations and Programs.

2. Individual interviews were conducted with the Dean of Curricula and selected curricular officers.

3. A general briefing was held for the Academic Dean, the Dean of Programs, the Dean of Research and Administration, and the Director of the Computer Center.

4. A general briefing was held for all academic department chairmen.

Many items were discussed at these meetings which greatly aided in the final resolution of output formats and file items for the new system.

G. PROBLEMS WITH THE ORIGINAL SYSTEM

Most of the problems with the original system stemmed from the lock-step processing procedure necessary to first obtain grades on all "5" cards before they could be sorted in student order. It was not possible under the present

system to obtain a listing of individual student registrations until after the grading process was completed! The most serious problems of the original system are listed below.

1. Changes to one card in any file usually required that similar information on other cards also be changed. The extreme case was a change of designator, which required every student "J" card and all previous "5" cards to be changed. This was necessary because the computer programs recognized students by a combination of file number and designator number.

2. There was a high probability that intermediate processing cards would be mislaid during card operations. For example, a recent Dean's list was produced without a small group of "H" cards. When the error was discovered, tedious checking of names produced another run of the Dean's list that was satisfactory but the statistics for frequency distribution were not recovered.

3. There was not adequate feedback of changes made to items in the files to recover from changes made in error. For each quarter there was a significant number of students registered erroneously in courses. These errors were not discovered until the instructor failed to report a grade for the student. In other words, the student had no way of knowing in which classes he had been registered until the Registrar or his curricular officer contacted him after the quarter ended regarding an unusual situation. Another common

problem was for an instructor to neglect to register some students in "special studies" classes until after the quarter is over. This and similar problems could be alleviated by the production of a student verification list of current registration.

4. The curricular officer had no official listing of where his students had registered themselves. In some cases, due to misunderstandings and in the case of foreign students, language difficulty, this has resulted in students attending the wrong classes for many weeks. It was theoretically possible for a student to change segments (hours of class meeting) without the approval or knowledge of his curricular officer. A listing given to the curricular officer by student name and course segment would have other uses besides aiding in the location of students for emergencies.

5. Because of the high priority of processing the graduating students, the production of the Dean's List was significantly delayed. A student who received his Dean's List letter in the fourth week of the new quarter was not well impressed with the data processing performance of the school.

6. The processing of the academic record for a potential graduate without color-coded cards--which occurred when a student's name was added late to the graduate list--was subject to error. For example, an additional card was sometimes found mixed in with the regular on-board student cards a few hours after graduating student academic records had

been taken to the computer center. If this student was a "doubtful graduate," the result could be embarrassing should recovery of such stray cards not occur in time.

7. The recomputation of quality point average and total credits passed when a student retakes a course due to a "D" grade, relied on the curricular officer notifying the Registrar of this fact. There was no way of signalling duplicate courses taken for credit other than visual monitoring.

8. If a few professors failed to turn in their grades at the appointed hour, the production of graduates' (and all other students') grades was seriously delayed. This has resulted in the extreme case of the Registrar's awarding an administrative Incomplete to a significant number of non-graduating students in order to start the production of graduates' academic records.

9. From a planner's point of view, the original system made any search of the files practically unworkable. If the answer to a question could not be found by hand search, then the next best solution would have been to place the whole card file on magnetic tape and write a special-purpose computer program to extract and sort the needed data. This second alternative was so cumbersome that it had never been attempted.

III. GOALS FOR THE NEW SYSTEM

As a result of the analysis of the present system and discussions held with the Registrar and Dean of Curricula, goals for a new system were established. These goals were used in subsequent design phases to resolve areas of difficulty, especially in the design of new output formats.

A. TIMING CONSIDERATIONS

A system goal was established to produce and distribute final class rosters to professors, academic records to curricular officers, and grade reports to students within twenty-four hours after grades had been submitted by professors. If grade submission were staggered, this would result in a grade report to the student twenty-four hours after the last grade for him had been submitted to the Registrar's Office. In order to meet this goal, a new prepunched grade-reporting card was designed and tested in parallel with the present grade-reporting system during the grading period at the end of Quarter 702. Professors were asked to submit grades both by marking the new grade card (Figure 2) and by marking the present system's grade roster. Comments regarding the new method were also solicited on a covering sheet forwarded to the professors with the grade cards and grade rosters. Of 260 sheets forwarded, comments were returned by nine instructors. All but one of these sheets were generally favorable to the new grading method.

Figure 2. New Grade Card.

One professor mildly objected to the new system, saying that there is a greater chance of error on the part of the professor when he transfers his grades from the roster worksheet to the dissimilar punched card. Three professors objected to initialling each card; two stated a desire to continue sending rosters along with the grade cards; and one instructor wanted the grade deadline to be made more flexible in reporting grades for those students not graduating or whose grades are not urgently needed by their curricular officer.

The purpose of the initial on each grade card was for resolving potential disputes in the future about the originator of a challenged grade card. Each grade card was assigned an input sequence number by the updating program, and this number stored in the registration record along with the grade assigned. Quick retrieval of the actual card, in cases of dispute, is assured.

The processing for the new grade cards was designed to eliminate the keypunching of grades, which was a bottleneck in the original card system. After the new grade cards are returned from the professors with the grade written in the target area, they will be quickly sorted manually into baskets that correspond to the legal marking grades: A, B, C, D, X, and I (for incomplete). The cards in the various baskets will be removed periodically, verified by eye-scanning the target area (all A's, B's, etc.) and placed in the input trays to the computer run with applicable header cards for each grade group. No specific ordering by students

or course is required because the elements creating the key for direct lookup of the specific registration record will be pre-punched into the card. As the cards are read into the computer they will be assigned a unique sequence number which corresponds to the position of the card in the input sequence. This sequence number is recorded along with the mark for the course in the disk file registration record. The card number appears on the regular input listing and on the special grade listing used by the Registrar during this period to answer professors' questions as to the status of grades already submitted. The card number can be used at any time to quickly locate a specific card because the input decks are normally not disbanded after a run is completed. Safeguards for missing, duplicate, and illegal grade cards will be built into the input processing program. A special control listing showing those courses with graduates and "doubtful graduates" for whom no grades have been received is also called for periodically during this period to combat the problem of late submission by some professors.

Plans for the new system include sending new class rosters each time any change occurs to the course or registration records; thus, each professor should receive multiple copies of the roster by the end of each quarter. The professor will continue to receive final (official) class rosters (with printed grades) within twenty-four hours after grades are submitted.

The flexibility of the grade reporting deadline is a policy matter involving the Dean of Curricula as well as the Registrar. However, curricular officers interviewed for this study indicated a need for quicker receipt of grades than the original system made possible. The new system should solve this problem for the curricular officer, and quicker turnaround should alleviate some of the professors' objections.

B. RELIABILITY OF DATA

In order to decrease the possibility of file updates made in error, a goal of receipt memoranda for all changes made to any file was adopted. For example, when an approved change-of-registration form is received by the Registrar, the file-updating program of the new system would automatically produce a memorandum to the instructor stating who had been dropped (or added) to which of his course segments. A courtesy copy of a new class roster for the applicable course segment would be sent along. Also, the student would be informed automatically, and a revised student verification sheet would be produced which would show his revised registration schedule. The curricular officer concerned would also be informed by the production of a small insert for his student verification listing. If the number of changes should become unmanageable to the curricular officer, a complete new registration listing for all his students could be requested from the Registrar.

The need for verification of a student's registration and miscellaneous information kept on file for him became apparent from the problems discovered with the original system. The goal was adopted that each student would receive a verification sheet in his student mail box once each quarter. This sheet should be produced in the third or fourth week after the professor had a chance to correct the preliminary class rosters. Any changes to items on the student verification sheet would be forwarded to the Registrar via the curricular officer so that he could verify the student's suggested changes. Both the student and the curricular officer would receive notification when the changes suggested had been placed in the record. Rapid turnaround for file updates is essential if the new system is to achieve the confidence and participation of students, faculty, and administrators. For this reason daily or triweekly file update runs should be planned as a goal of the system.

C. SECURITY AND DISASTER RECOVERABILITY

The new system was designed with these security and disaster recoverability goals:

1. The storage medium would be portable 2311 disk units with a storage capacity of 7.25 million bytes (characters) per unit.

2. The basic processing cycle would be from a "father" disk to a "son" disk. All updating would be performed on the son disk so that in case of machine or program malfunction,

the father disk would remain unchanged and could be used as a basis for the next run (or rerun).

3. After a run is completed, the father disk and the son disk would be stored under lock and key in separate buildings. This would have the dual advantage of discouraging tampering and make recoverability of data feasible following a major disaster to one of the buildings.

4. As a further backup to the remote possibility that both the father and son disks be destroyed during updating, either a "grandfather" disk should be saved or periodic dumps of the Registrar's disk files to magnetic tape should be planned.

5. The use of the OS/360 operating system password capability should be instituted as soon as the academic files are established. This would improve the security of the disks by discouraging unauthorized access to the academic record files for the short time that they are mounted. Whereas the use of the password system does not by itself insure security, it would bring to the computer operators' attention the fact that the Registrar's files require special operating procedures and safeguards.[6] The portability of the disk packs provides many advantages over the cumbersome card files. However, the extremely high data-packing density of the disk files and the ease of accessibility to them requires the additional measures outlined above.

---

[6]For a description of the job control language of the password system, see Gary Deward Brown, System/360 Job Control Language, pp. 236-237.

D. EASE OF INPUT-OUTPUT BATCH PROCESSING

Besides eliminating the bottleneck at grade-reporting time discussed above in conjunction with the new grade card, there were other features incorporated in the design goals regarding inputs and outputs.

1. Make Input to Files Easier

The files were designed so that each data entity appears only once in the whole file system. For example a change to a student name will only have to be made in one item of the student record. All other references to the student name from any other part of the files will refer to this item. The hardware item that makes this possible is the direct-access capability of the disk storage units. The software complement to this capability is the extensive use of pointers as relations between data items in different files. This scheme of single-datum items and pointers is in marked contrast to the present punched-card system that incorporated much redundant data between cards in a file and which was necessary to keep the punched-card files working together.

Because updates to the disk files will affect many files, an elaborate system of pre-checking by the computer program was instituted in order to prevent obvious errors from being introduced into the disk files. A separate listing of input errors was developed to make error correction easier for the worker responsible for this vital function.

2. Make Delivery of Output Easier

The output formats were designed with the goal of eliminating the large amount of form-bursting and form-sorting time necessary for output forms under the present system. Also because many disparate forms will be produced in each computer run under the new system, it was necessary to standardize the output formats to make quick delivery possible. Output forms will be prcduced in the following batches:

a. Wide forms used by the Registrar as unburst listings (school-wide registration lists, input card lists, error lists, etc.).

b. Wide forms used by curricular officers and others in the form of special searches of the files.

c. Narrow forms cut to size in one batch at the print shop:

(1) Student forms in school-wide alphabetical order (academic records for Registrar's files).

(2) Class rosters and roster changes in department/instructor code order for delivery via the inter-school mail.

(3) Curricular officer lists of registrations and academic records in either alphabetical order by curricular officer code or by section order within curricular officer code.[7]

---

[7]Each curricular officer will have a choice of the order in which he receives his output. This choice will correspond to the order in which each curricular officer maintains his student files.

(4) Student forms (grade reports and verification sheets) in student mail center box number order for quick delivery to students.

All of the curricular officers interviewed for this study agreed that the use of the Student Mail Center would speed up the delivery of forms from the Registrar to the student. The use of pre-printed forms did not fall into the regular operating procedures of the computer center. As the output formats were developed, it became clear that the capability to easily change the descriptive information on all forms was a definite asset. Also, the increased turnaround time at the computer center when pre-printed forms are used, was considered as a factor against the use of these forms.

3. Flexibility in Scheduling Outputs

The exact time for the production of each output varies from quarter to quarter, depending on the back-to-back schedule and also on the status of the inputs available at any one point in time. It was necessary, therefore, to establish a flexible method of calling for various reports other than those produced automatically in the process of updating the files. This flexibility also permitted the re-running of any reports because of input data problems or for any other reason.

E. SYSTEM QUERY CAPABILITY

One of the serious problems with the original system was that it could not be made to easily respond to questions not

previously programmed or thought of in advance. The design of the query system for the new system met the following goals:

1. Records could be selected from the various files on disk by the use of simple logical expressions. For example, the search for all Supply Corps officers enrolled in the Management or Computer Systems Management curricula would be expressed as follows:

```
(SELECT: STUDENT RECORDS) (DESIGNATOR = 31XX OR 37XX)
      (AND) (EDCODE = 8159 OR 8173)
```

2. Records, once selected, should be used to produce output in one of many forms: counts, printed lists, punched cards, magnetic tape, or other data sets.

3. The lists or cards should be produced in the order of one of several built-in sort chains. For student records this would be one of the following:

a. Alphabetical order

b. Curricular officer code/alphabetical order

c. Curricular officer code/section/alphabetical order

d. Record number order

e. Student mail center order

For course records the order would be one of the following:

a. Record number order (quarter/course number/ segment number)

b. Professor order (academic department/quarter/ instructor code/course number).

4. For printed lists, the titles and the file items to be printed should be specified by the user. The capability for placing specific titles (and items) in specific printer columns (1-132) should also be provided.

5. For punch cards, magnetic tape, or other data sets, the capability for providing specific file items in specific character positions (1-80) should be provided.

6. For counts the title of the count should be provided by the user.

The design of the query system was purposely kept simple so that some experience could be gained with satisfying search requests. In the future a more elaborate data retrieval system will probably be required.[8]

F. TABLE AND OUTPUT DATA CHANGES

It was anticipated that the following table information would be needed for processing the Registrar's files:

	Table	Estimated No. of Entries
1.	Academic Calendar	9
2.	Education Code	59
3.	University Code (Entrance Credits)	989
4.	Country Code (Foreign Officers)	30

Each of these tables is subject to frequent change, the university codes increasing at the rate of about thirty new

---

[8]For a description of a more elaborate, but practical system, see A. L. Humphrey and W. G. Munro, "Management Information Retrieval" in The Computer Journal, Vol. 13, No. 2, May 1970, p. 127-130.

codes per quarter. It was adopted as a goal of the new system that there be an easy method for updating and printing the status of all table information. The establishment of a separate table file on the disk unit made the addition of paragraph information for each printed output a natural adaptation of the table concept. As an example of this, the number of table line entries necessary for all paragraph information on the student verification sheet was only twenty-five. It then became a relatively simple matter to update the verbiage of this report by table update functions instead of requiring specific program changes. The table concept also allows an easy trade-off between maintaining some (or all) tables in the computer central core during execution or maintaining the tables on the disk at the expense of slightly greater processing time.[9]

G. TREATMENT OF GRADING SYSTEM CHANGES

The grading system at the Naval Postgraduate School has remained stable for the past thirty-five years with the following academic marking system:

---

[9]Because the table information is blocked (45 records per block), the core access of 3500 characters will require only one or two disk accesses.

Mark	Performance	Quality Points
A	Excellent	3
B	Good	2
C	Fair	1
D	Barely Passing	0
X	Failing	-1
WP	Withdrew Passing	0
WX	Withdrew Failing	-1

Prior to 1935-1936 the following marks were used:

Mark	Performance[10]
E	Excellent
G	Good
P	Passing, But Not Too Thorough
X	Unsatisfactory

The possibility of changing the grading system has been raised by the Faculty Committee on Scholastic Standards and Honors. A goal for the new system was to be able to cope with changing grading systems. When a grading change is made, the new system must be able to convert two systems (at least) of grades to numeric quality points. The method for checking for legal grades and computing their numeric equivalents utilized a table similar to the following.

First Quarter	Last Quarter	Legal Grade (3 Characters)	Numeric Equivalent
701	704	A	3.00
701		B	2.00
701		C	1.00
701		D	0.00
701		X	-1.00
701		W/P	0.00
701		W/F	-1.00
701		I	0.00
704		A-	2.80
705		A	2.95
706		NUM	*

[10]Naval Postgraduate School, "Office of the Registrar--Transcript Information" (form attached to all official transcripts issued by the school).

This table is a hypothetical one, but it shows what could be accomplished within a changing grade environment. In the table above the present grading scheme runs from Quarter 701 through Quarter 704. During and after Quarter 704, the grade A- is recognized as a legal grade with a numeric equivalent of 2.80; the straight A grade continues to have a value of 3.00 during Quarter 704 but is changed to 2.95 during subsequent quarters. During Quarter 705 a three-digit numeric value is also accepted as a legal grade and is its own numeric equivalent. The letter grades and A- continue to be accepted during Quarter 705.

H. HISTORICAL RECORD AND HISTORICAL DATA BASE QUERY CAPABILITY

The academic data-base files for the new system were planned to accept data starting with the first quarter of Academic Year 1970-1971 and build continuously from that point in time. Because the present system maintains records for only one academic year at a time, it was felt that the 1 July 1970 as-of date for the new file would be a convenient starting point. Parallel operation with the present system would continue until the data base files were loaded and the processing programs debugged. There will come a convenient time for purging the system of information on students no longer on board. Without a purge process the files will grow without bound and will not be able to be contained on one 2311 disk unit. The problem of historical

records and searches thereto was planned from the very start of the analysis.

The purge process is scheduled to be performed yearly in September after the grade distribution report is prepared in August. The end of August becomes a convenient time to use as a benchmark for saving historical records. At the end of August of each year a complete dump of the disk files should be made to magnetic tape. Because the processing programs also reside on the disk, they too will be placed on the magnetic tape. It is then possible at any time in the future to use the utility restore program to place a new disk in the exact condition as was the disk at the end of August. Because the computer programs will also be restored to disk, it will then be possible to use the query system to perform searches of the data base "as of" August of the year in question. After a number of years have elapsed and a number of historical tapes have been created, it will be possible to perform similar searches on successive yearly files to obtain comparison data.

This historical record system may appear to be deceptively simple, but it has a number of advantages:

1. Changes to items in the data base must always be accompanied by changes in the processing programs, updating routines, and query system. Because the programs reside with the data base they were designed to service, they should never get out of step from an historical-search point

of view. All that need be retained is a user's guide to successive versions of the query system.

2. The alternative method is to keep two systems running simultaneously: one for the current operating system and another for historical records. Inevitably, the historical system would not be kept up-to-date. When new items are added to the data base for pressing operational reasons, there would be no convenient way of making the records compatible with the historical file.

3. Assuredly, changes will occur in the academic data base items and processing method. Record lengths will change, record key compositions will be altered, and the number of items in the files will change. In short, the program maintenance effort alone indicated that the first method was the better method, at least until a great number of historical records have been collected and some experience gained as to the type of historical searches requested.

I. FUTURE SYSTEM INTERFACES

Three future interface areas were considered during the design phase of the new system. These interfaces were the student data bank, the automated student scheduling system, and the longer-range planning information system.

1. The Student Data Bank

A capability for cross-verification runs with the student data bank was established as a goal of the new system. The exact timing and form of this cross-verification run will have to wait until both systems are in full

operation. The Registrar's data bank has purposely been limited in scope to items of academic interest and isolated from general access methods for reasons of security and data reliability. Whereas there are similar problems with student data bank items, they probably will not be so acute.

2. The Automated Student Scheduling System

The efficient and timely scheduling of students into class segments has been a long-sought-after goal of the Naval Postgraduate School. The development of another computer system to accomplish this scheduling proceeded in parallel with this study, and both systems were kept compatible. A system goal was established to accept scheduling information on all students prior to the beginning of the quarter by direct data transfer instead of by course cards after the quarter has begun. It would then be possible to publish class rosters for instructors and registration information for students and curricular officers prior to the beginning of classes each quarter. The following data items (currently unused) have been established for each course segment:

a. Type of meeting code (1 = lecture, 2 = lab, 3 = other).

b. Day of the week (example: 'MTW F')

c. Hour of the day (example: 08 means 0800)

d. Length in hours

e. Room utilized

These items, once established by the scheduling system, would be used to publish meaningful class rosters

and registration schedules. They would be updated by the same means as the other academic data items:

a. By the instructor verifying the initial class roster

b. By the student verification sheet

c. By changes received from other official sources.

3. The Planning Information System

Once the files of the academic data base are established and are being verified by the numerous feedback devices planned, the information should be reliable and should be used for a number of planning functions. Some of the more obvious applications are:

a. Actual room utilization figures by number of students and hours of the day

b. Instructor contact hour studies

c. Student program cost studies

IV. HOW THE NEW SYSTEM DIFFERS FROM THE ORIGINAL SYSTEM

Extensive analysis of the original system altered the format of some of the original system outputs. One output, the class roster letter, was marked for possible elimination after a suitable trial period with the new system. Several new inputs and outputs were planned, and these are discussed below.

A. CHANGES MADE THAT WERE COMMON TO ALL REPORTS

The following changes were made in the interest of consistency for all reports produced by the Registrar:

1. All sheets list the originator of the report as "The Registrar, Naval Postgraduate School," or the shortened form, Code 0221.

2. For reports that have more than one page, the page number occurs at the top of the second and successive pages.

3. The name of the report and the "as of" or run date appear at the top of every page.

4. For reports delivered by the inter-school mail system or by the student mail center, a large-block-character generator was used to aid in the delivery process.

Figure 3 is an example of the student verification sheet format.

```
 SSSSS   M     M   CCCCC          22222    33333      1      88888
S     S  MM   MM  C     C        2     2  3     3    1     8     8
S        M M M M  C                    2        3    1     8     8
 SSSSS   M  M  M  C               2222       333     1      88888
      S  M     M  C              2              3    1     8     8
S     S  M     M  C     C        2        3     3    1     8     8
 SSSSS   M     M   CCCCC         2222222   33333   1111111   88888
```

FROM: THE REGISTRAR, NAVAL POSTGRADUATE SCHOOL (CODE 0221)

TO: LT MARVIN R. AARDAL, USNR, 632329/1100 (SMC 2318)

SUBJ: ACADEMIC DATA BANK VERIFICATION (24 MAR 70)

1. THE FOLLOWING IS A LISTING OF SELECTED ITEMS MAINTAINED IN THE ACADEMIC DATA BANK. PLEASE VERIFY THE LISTING AND FORWARD ONLY SHEETS WITH INCORRECT ITEMS (CIRCLED AND CORRECTED) TO THE REGISTRAR VIA YOUR CURRICULAR OFFICER. CHANGES IN GRADES OR CLASS REGISTRATION MUST BE ACCOMPLISHED BY APPROPRIATE CHANGE FORMS.

2. PRESENT CLASS REGISTRATION:

COURSE	CATALOG/SEG	HOURS R	HOURS L	PROFESSOR
INTRO TO COMP & PROGMG	CS2100-11	4	0	SINGER, E. A.
SYS ANALYSIS I	OA3611-3	4	0	MARSHALL, K. T.
SYS SIMULATION	OA3653-10	4	0	READ, R. R.
UNDERWATER ACOUSTICS	PH3421-2	4	2	GARRETTSON, G. A.

3. ADMINISTRATIVE ITEMS:

A. OBLISERV START DATE: 0769
B. ESTIMATED GRAD DATE: 0371
C. STUDENT SECTION NO: CS02
D. OFFICER RANK: LCDR
E. CORPS OR COUNTRY: USNR
F. DESIGNATOR: 1100
G. QPA LAST QUARTER: 2.00 (GRAD) 2.00 (TOTAL)
H. QPA CUMULATIVE: 2.04 (GRAD) 1.92 (TOTAL)
I. ENTRANCE CREDIT: HARVARD AB 1957
J. CURRICULUM NAME: OPERATIONS ANALYSIS
K. SOCIAL SEC. NO.: 550-48-2052
L. FILE NUMBER: 632329

4. SPECIAL NOTES:

A. YOUR COLLEGE TRANSCRIPT IS NOT ON FILE. PLEASE CONTACT THE REGISTRAR.

FIRST ENDORSEMENT (FOR INCORRECT ITEMS ONLY) DATE_____
FROM: LT MARVIN R. AARDAL, USNR, 632329/1100 (SMC 2318)
TO: THE REGISTRAR, NAVAL POSTGRADUATE SCHOOL (CODE 0221)
VIA: CURRICULAR OFFICER, NAVAL MANAGEMENT (CODE 36)
1. RETURNED WITH INCORRECT ITEMS CIRCLED AND CORRECTED.

SIGNED:_____

SECOND ENDORSEMENT
FROM: CURRICULAR OFFICER, NAVAL MANAGEMENT (CODE 36)
TO: THE REGISTRAR, NAVAL POSTGRADUATE SCHOOL (CODE 0221)
1. INCORRECT ITEMS NOTED AND VERIFIED AS CORRECT.

SIGNED:_____

FIGURE 3 - STUDENT VERIFICATION SHEET

B. NEW SYSTEM OUTPUTS

The following is a list of new outputs:

1. Textbook allowance checklist

2. Current registration lists for Registrar and curricular officers

3. Quality point average summary for curricular officers

4. Instructor academic load report for the Dean of Programs

5. Student grade report (replaces the third copy of academic record)

6. Thesis title list.

C. NEW SYSTEM VERIFICATION DEVICES

The following is a list of new verification devices distributed outside of the Registrar's office:

1. Student verification sheet (see Figure 3)

2. Change of registration memorandum to curricular officers and instructors

3. Change of grade memorandum to curricular officers and instructors

The following is a list of new control lists used within the Registrar's office:

1. List of grades processed this quarter

2. List of incomplete grades

3. List of critical grades missing

4. Course control sheet

The following is a list of control decks that can be called for use in the Registrar's office:

1. Prospective input students
2. Onboard students
3. Prospective departures

In addition to the above a standard count of records is tabulated during each computer run and printed at the end of run showing the following mutually exclusive totals:

	USN	USMC	USCG	ARMY	A/F	FOREIGN	STAFF	AVSAFE	TOTAL
Pros. Input									
Onboard									
Prospective grad									
Continuing									
Not Onboard									
Total Records									

D. NEW SYSTEM INPUTS

The following new data items were necessary to order to produce the outputs:

1. Social Security Number

The Bureau of Naval Personnel has directed that all personnel records for naval personnel be converted to the use of social security number instead of file number by 1 January 1972.[11] Accordingly, an item in the student master

---

[11] BuPers Instruction 1070.20 series.

record has been reserved for nine digits of social security number; and plans have been formulated for the one-time gathering of this item on all past students. The continuing process of obtaining this item on all new students has already been started. All report formats were designed so that when a master suppress-file-number switch is set, the social security number will appear in the place of the present file number. If this switch is not set, both the social security number and the file number will appear on the student verification sheet.

2. Student Mail Center Box Number

This item for all on-board students was obtained by a one-time computer program which collated the information available in the military personnel office with the names in the master student file. For all new students this item has been obtained on the Registrar's information sheet filled out by the student upon reporting for duty.

3. Foreign Officer Corps Code

Not all foreign officer students are members of the naval establishment of their countries, and this has been a source of difficulty in the past. Accordingly, the present arrangement of a combined corps and country code has been split into two items, officer corps abbreviation and country code. For U.S. officers the corps abbreviation contains USN, USNR, USAF, US(MR), etc. For foreign officers the corps abbreviation contains NAVY except for some cases where AF, MSDF, etc., is appropriate. The Country table

stored on disk contains both the noun and adjective forms for foreign country codes. It then becomes possible to print name salutations such as the following:

LT John D. Adams, USCG

LT Antonio Almeida, Portuguese Navy

LT Angelos Argyropoulos, Greek Navy

MAJ Aharon Beth-Halachmi, Israeli AF

E. COMPUTER RUN PHILOSOPHY FOR THE NEW SYSTEM

Inputs for each computer run under the new system will be in the form of punched cards. Input cards can be submitted in the input deck in any order. The functions performed by the computer run will be designated by the use of input header cards, and the data to be changed will be in the form of detail cards. Header cards are divided into four types. They are identified by a keyword which starts in the first column of the card.

Keyword Identifier	Header Card Purposes
UPDATE	Initiates update functions for files
REPORT	Sets up a specific report call
PURGE	Initiates system purge
QUERY	Activates the query system for one search

Parameters for header cards are enclosed within parentheses and are not restricted to specific card columns. The overall input philosophy for computer runs is that cards should be punched and inserted into the input tray as events occur requiring them. If a special report is required, the

header card calling for that report is inserted in the input tray in back of the existing cards.[12] For each specific search a set of control cards activating the query system is prepared and placed in back of the existing input cards. In other words, the Registrar's normal processing should be to take care of business transaction "as occurring" and the input card deck built up in the same order. Periodically (three times per week or perhaps more often during peak processing times), the input tray is taken to the computer center along with the applicable disk units for a regular computer run. Meanwhile, a new input tray is started for new transactions. If a computer run is not successful, the input cards for that run are combined with the accumulating new cards and another computer run is attempted.

Once the outputs from a computer run are satisfactory, the input card deck for the successful run should be stored and not disturbed except in unusual circumstances. The input card checking routines were designed so that cards that have obvious problems will be returned in the form of an exact duplicate of the original input card. This "oops" output card can be corrected (in most cases) and submitted in a later input deck. By storing successive generations of input cards along with the disks that were used in

---

[12]The sequence of computer program procedures is such that all file updates are accomplished before any reports or special searches are performed. This means that all reports and searches have the most recent file information available.

processing them, recovery from major and minor disasters becomes a realistic possibility. There are obvious upper limits on the number of generations that need to be stored (different for input decks and disks), and this whole operation should be coordinated with the annual historical tape dump. The optimum disk cycle schedule developed will be a function of the following variables:

1. Number of reruns necessary because of machine errors
2. Number of reruns necessary because of input problems
3. Number of disk packs made available for Registrar's use
4. Frequency of computer runs made
5. Degree of backup safety desired
6. Frequency of disk-to-tape dumps made.

V. PROGRAMMING LANGUAGE SELECTION AND PROGRAMMING CONVENTIONS USED

The selection of a programming language was made after the goals for the new system were established. The establishment of files, file names, file items, and file item names was accomplished using conventions given below.

A. LANGUAGE SELECTION

The selection of a programming language was made after the goals for the new system were established. The following criteria were among those considered:

1. Large number of files and file manipulations required
2. Ease of programming direct-access disk functions
3. Ease of reprogramming and making changes to programs already written
4. Documentation features of languages
5. Probability of continued language support by the computer center during all hours of the day
6. Programming language efficiency

The language selection was quickly narrowed to a choice between COBOL and PL/1. PL/1 was selected because at the time the language decision was made, the computer center was considering expanding their time-sharing system to absorb a considerable portion of the working day and thereby precluded COBOL support during these hours. PL/1 support was assured for round-the-clock operations and was also supported in

both batch and terminal modes under the time-sharing system. COBOL is not supported by any known time-sharing system; and in particular it was not supported by the three systems under active consideration by the computer center staff.

The decision to program the new system for batch operation (at least initially) was made for the following reasons:

1. The final selection of a specific time-sharing system was in great doubt.

2. There is not a great deal of experience to draw upon in the use of the indexed-sequential access method in a multi-file application under a time-shared environment;

3. The immediate access capability of a terminal system is not absolutely essential to the Registrar's needs. The capability to obtain batch turnaround of a few hours (in an emergency) is all that was absolutely required.

4. The problem of security of file access, while not an impossible or unworkable problem under a terminal mode, would be much more complicated than under a batch system setup.

Once the Registrar's files are established and are being updated by the batch system outlined in this study, it should be a relatively easy matter to program an independent query system in the future to access the data files should the need for this type of response become apparent.

B. PROGRAMMING CONVENTIONS USED

1. Variable Names

The establishment of files, file names, file items, file item names, and other variable names were accomplished with these criteria in mind:

a. Consistency: Consistently-named variables help in avoiding programming traps and make the reading of programs by others easier.

b. Readability: Variable names should bear some relation to the function they serve.

c. Hierarchy: Variables for file items should identify the file in which they originate.

d. Type: Variable types should be recognizable from the file name:

Variable Use	Variable Name	Variable Name
File name	PFS	Professor file (son)
File item	P_FLAG	Prof. file delete flag
File key item	P_REC #	Prof. file record No.
File pointer item	P_LP DEPT	Prof. file left pointer, dept. chain
Based variable pointer	S	Student structure pointer
Index variable	J	Integer index

2. Other Conventions Used

The following is a list of other conventions used:

a. Program blocks were indented five spaces for each sublevel to indicate nesting levels.

b. Descriptive comments were inserted into the programs as they were written to improve readability and ease the problems of future changes.

c. Wherever possible the use of STATIC storage class was used to reduce the size of the compiled program.

d. Unusual ends to program blocks were ended with a call to an error routine with the reason for the "error" incorporated into the routine call. This procedure helped to identify unusual or unplanned-for situations and keeps the processing safely moving along.

C. MULTI-TASKING AND OVERLAY CAPABILITY

The initial design of the processing program divided the program into these segments:

1. Initialization segment for global variables
2. Processing of input cards
3. Alphabetic chain segment (output reports)
4. Organization code chain segment (output reports)
5. Student mail center chain segment (output reports).

The programming accomplished to prove the feasibility of the system showed that segment 2 would be the largest programming segment. This segment must also be completed before segments 3, 4, and 5 can be commenced. The possibility of using the program overlay features of OS/360 should be considered after the rest of the procedure program sizes are known. It may be possible that segments 3, 4, and 5 may all be able to reside in the same area as segment 2. Multi-tasking should also be considered because each of segments

3, 4, and 5 can be performed independently of the others, at least theoretically. However, disk-file-head interference may demonstrate that the multi-tasking setup is not efficient.

VI. DISK FILE ORGANIZATION AND INITIAL LOADING OF THE ACADEMIC DATA BANK

The ten disk files of the academic base were designed to store information under the following criteria:

1. Ability to produce the outputs of the new system
2. Ability to respond to future data retrieval needs
3. Ability to be expanded easily as the need for new data items become apparent.

Extensive list-processing pointers were used to tie the various files together. At the time of this writing the final loading of the files was underway. The initial loading must be completed before a great deal of useful work from the files can be accomplished.

A. DISK FILES AND THEIR RELATIONS

The following is a list of the 2311 disk files set up in the Registrar's data bank:

Item	File	Record Length (bytes)	No. CYL Alloc.	Per Cent Total No. CYL
1	Master record file	2084	1	0.5
2	Student file	544	50	25.0
3	Student index file	32	10	5.0
4	Professor file	240	8	4.0
5	Professor index file	27	2	1.0
6	Course file	172	12	6.0
7	Registration file	55	44	22.0
8	Thesis title file	301	5	2.5
9	Entrance credit file	22	4	2.0
10	Table file	78	7	3.5
	Total		143	71.5

The files, the file items, their index keys, and the relations between files are described below.

1. The Master Record File

This file consists of 252 control items, arranged logically in five groups, but physically in one blocked record of 2084 bytes. This record contains the master items for the rest of the data bank. During processing the master items remain in core but are rewritten on disk any time one of the master items is changed. This will occur only when new items are added to any of the files or when report calls are interpreted in the input deck. This file is a sequential file of one record only and has no key.

a. First group--master keys

(1) Master run number

(2) Student record number of top of SMC chain

(3) Student record number of top of non-SMC chain

(4) Professor record number of top of alphabetical professor chain

(5) Student record number of top of student alphabetical chain

b. Second group--curricular officer items (items repeated for each curricular officer)

(1) Organization code number

(2) Output order choice bit

(3) Student record number for top of CURR/ALPHA chain

(4) Student record number for top of CURR/SECTION/ALPHA chain

(5) Curricular officer title

(6) Eight spare switches

c. Third group--academic department items (items repeated for each academic department)

(1) Organization code number

(2) Professor record number for top of DEPT/PROF chain

(3) Eight spare switches

(4) One spare character string

(5) Department title

d. Fourth group--numeric values

(1) Last student record number used

(2) Last professor record number used

(3) Last thesis record number used

(4) Five spare numeric variables

e. Fifth group--switches

(1) Call to produce all class rosters

(2) Call to produce all verification sheets

(3) Call to produce the roster letters

(4) Call to produce textbook checklist for next quarter

(5) Call to produce a supplementary textbook checklist

(6) Call to produce the grade cards

(7) Call to produce the course cards for next quarter

(8) Call to produce the Dean's List

(9) Switch to suppress all file number information

(10) Switch to suppress room number information.

2. The Student File

This file consists of 544 characters and is an indexed sequential file, which means that it can be accessed either sequentially by increasing key sequence, or directly if the key is known. Because the key is used extensively as pointers in other files and as part of the key for other files, it was mandatory to use as compact a key as possible for this file. The name field, while distinct, was too long. The officer file number was deemed unsatisfactory because it is being discontinued shortly, the social security number was unsatisfactory because it was not immediately available for all students, and in the case of foreign students would have to be created. Therefore, an internally generated student record number was used. This number is assigned by the computer program when the student record is initially set up. The student index file (see below) serves the process of converting (with one disk seek, usually) the name character string to the assigned numeric record number. To increase processing speed the student record number will be placed on the course and grade cards (also produced by the computer program), thus bypassing the conversion step. For external queries and updates it is never mandatory to know a student record number--name alone is sufficient.

The possibility of two students having the same name character string is reduced by checking for existing duplicate student names at the point a new student record is submitted for insertion in the file. At that point in time an error message is created indicating that the duplicate name problem exists. As a result one (or both) student names should be altered so that they are made distinct. The insertion of full middle names, Jr., etc. are obvious remedies to keep the subsequent processing of the two students' records from affecting each other. This problem has school-wide implications and should be considered on an as-occurring basis.

The student record items are grouped logically into seven groups:

a. First group--identifier items

(1) Record delete flag

(2) Record number (this is the embedded key for the record)

(3) Student name (last name, first name, and initial(s)

(4) Rank

(5) Section

(6) Corps

(7) Country code

(8) Serial number

(9) Social security number

(10) Obligated service start date (matriculation date)

(11) Estimated graduation date

(12) Designator

(13) Staff code (mail delivery code if no SMC)

b. Second group--status switches (bits)

(1) Transcript on file

(2) Prospective input

(3) On-board

(4) Other type student (other than a student officer)

(5) Civilian

(6) Aviation safety student

(7) Foreign student

(8) Doubtful graduate case

(9) Student continuing in another curriculum after graduating

(10) Five spare switches

c. Third group--numeric values

(1) Last quarter on Dean's List

(2) Second-to-last quarter on Dean's List

(3) Total number of times on Dean's List

(4) Last quarter for textbook checklist

(5) Second-to-last quarter for textbook checklist

(6) Two spare numeric variables

d. Fourth group--quality point average (QPA) group

(1) Quarter for which last QPA's computed

(2) Last quarter QPA (all courses)

(3) Last quarter QPA (graduate-level courses only

(4) Cumulative QPA (all courses in last curriculum for which grades have been received)

(5) Cumulative QPA (graduate-level courses only--last curriculum for which grades have been received)

e. Fifth group--pointer group

(1) Pointer to first registration record for latest quarter group

(2) Left pointer for SMC chain

(3) Right pointer for SMC chain

(4) Left pointer for alphabetical chain

(5) Right pointer for alphabetical chain

(6) Left pointer for curricular officer alphabetical chain

(7) Right pointer for curricular officer alphabetical chain

(8) Left pointer for curricular officer section chain

(9) Right pointer for curricular officer section chain

f. Sixth group--automatic report-generator group.

These items are set during the processing of input cards in segment 2 and are cleared as reports are produced in segments 3, 4, and 5.

(1) Verification-sheet-due-to-student character group. This is the completion to the sentence, "A new

verification sheet is forwarded to confirm changes made to your record in these items..."

(3) Grade change (three switches). These are indicators that at least one non-blank grade has been changed and that reports are called for.

(4) Grade change quarter

(5) Grade change characters. This is the completion of the sentence, "Academic marks have been changed since last printing for..."

(6) Registration change (three switches). These are used to indicate that change memoranda are due for this student.

(7) Registration change characters. This is the completion of the sentence, "Course registration has been changed since last printing for..."

g. Seventh group--curriculum header blocks. The following items are repeated five times for up to five different curricula for each student. There is one item to indicate how many curriculum blocks are active for this student.

(1) Quarter that this curriculum block starts

(2) Education code for this block. Curricular officer number and curriculum number are all derived from this code.

(3) Pointer to first registration record for this curriculum.

(4) Type of degree to be awarded

(5) Quarter in which degree was actually awarded

(6) Pointer to first thesis record number

(7) Two spare switches

(8) Credits brought forward--graduate credits attempted

(9) Credits brought forward--total credits attempted

(10) Credits brought forward--graduate credits passed

(11) Credits brought forward--total credits passed

(12) Credits brought forward--graduate quality points

(13) Credits brought forward--total quality points

The original design of the student record anticipated the use of variable-length records with the number of curricula blocks being the variable involved. However, it was discovered by programming test cases that the current version of the PL/1 compiler installed at the computer center does not support variable-length records for indexed sequential data sets. The next version of the compiler planned for installation will support variable length records, and a space savings of about 41 per cent in the length of the student record can be anticipated if this mode is adopted. Additional savings in the thesis record and the professor record can be gained by changing to variable-length

records when this mode is supported. In the meantime there is adequate room on one disk unit for all of the Registrar's files and programs and an adequate margin for item expansion over and above the spare items already built into the records.

3. The Student Index File

The purpose of the student index record is to provide a rapid means of converting a student name character field of twenty-two characters to the student record number. In those cases where no exact name match is found, this file rapidly provides a list of the ten most likely "near misses" which can be used by the keypunch operator in the Registrar's office to help resolve the name error. No processing is done on the student record until an exact name match is found. Because of this exacting requirement, it is anticipated that a large number of update records will be rejected when first submitted for update. The "near miss" capability has already proven to provide the correct name if that name resides in the master student file. See Section C, below, for a complete discussion of the hash-coding scheme used to create this index file and the results of trial runs using this scheme.

a. Item group for record

(1) Delete flag

(2) Compressed name. This is the embedded key which consists of the five-character hash code plus an additional character to make the key unique.

(3) Student name (twenty-two characters). This is used to obtain the exact match on name required before the student record number is recognized.

(4) Student record number.

4. The Professor File

There is one record for each instructor teaching classes at the Naval Postgraduate School. A numeric record number is assigned for each professor at the time the record is set up. There is a professor index file (see below) for converting from professor names to professor record numbers. Because the professor file is not as volatile as the student file in terms of number of records created per quarter, the involved hashing scheme used for the student index file is not used for the professor index, but a direct lookup on professor name is used to locate the record number. Again there is no need for the professor record number to be known outside of the computer program. In all cases the professor name is all that is required for updating purposes. The primary reason a professor number is used for internal processing is to conserve disk space. The professor record numbers are also used extensively as pointers in other files.

a. First group--identifier items

(1) Record delete flag

(2) Professor record number. This is the embedded key for the record.

(3) Professor name (twenty-three character consisting of last name and initials).

(4) Academic department organization number

(5) Two-letter mail delivery code

(6) Academic rank code

(7) Two spare character groups

(8) Eight spare switches

b. Second group--pointer group

(1) Left pointer for alphabetical chain (schoolwide)

(2) Right pointer for alphabetical chain (schoolwide)

(3) Left pointer for DEPT/ALPHA chain (department only)

(4) Right pointer for DEPT/ALPHA chain (department only)

(5) Two spare pointers

c. Third group--course header blocks

Each item is repeated eight times to yield access to the eight most recent quarters in which the instructor has been teaching course segments. There is one item to count how many blocks are active for this record.

(1) Quarter number

(2) Pointer to first course record for this quarter

(3) Spare switch

5. The Professor Index File

During the initial loading phase it was discovered that the punched-card files had coded some professors'

names with differing character strings. These character strings appear on the "A" cards and are used to print the professor name of class rosters. For example, the following strings all refer to Professor R. T. Williams:

WILLIAMS
WILLIAMS, R.
WILLIAMS, R. T.

The use of an index file was thus mandatory to resolve the professor names on existing records. Once all of the existing files are finally converted to the new data bank system, the duplicate entry names can be removed from the index file, and only full names with initials used in the future. At the time this report is being written, the professor index file consists of 603 records, whereas the professor file itself consists of 295 records.

a. Index items--only one group

(1) Record delete flag

(2) Professor name (twenty-three characters)

(3) Professor record number

6. The Course File

There is one course record of 172 bytes for each course segment taught at the school. The information in the file is common to all students registered in the course segment. Because course titles change from time to time, it was not possible to place this item in a higher-level file. During the conversion process one course file record was created for each "A" card in the registration files of the original punched-card system.

a. First group--identifier items

(1) Record delete flag

(2) Course record key. This is the embedded key consisting of eleven characters. For example, 701MA323202 where 701 = Academic quarter 1 of AY70-71

MA3232 = Course number

02 = Segment number

(3) Course title

(4) Primary professor record number

(5) Alternate professor record number. Some courses have a second professor assigned.

(6) Number of lecture hours of credit for course

(7) Number of laboratory hours of credit for course

(8) Three spare switches

b. Second group--pointers

(1) Left pointer, primary professor chain

(2) Right pointer, primary professor chain

(3) Left pointer, alternate professor chain

(4) Right pointer, alternate professor chain

(5) First pointer to top of student registration chain (points to first registraiton record for this course).

c. Third group--course location blocks (currently not used). Each item is repeated three times for up to three separate locations for this course segment.

(1) Type of meeting (1 = lecture, 2 = lab, 3 = other. This code can be expanded to account for other types.)

(2) Days of the week. Example: MTW F means Monday, Tuesday, Wednesday, Friday; M WTF means Monday, Wednesday, Thursday, Friday.

(3) Hour of the day. Example: 08 means the class convenes at 0800

(4) Number of hours (usually one)

(5) Room used. Example: H221 means Herrmann Hall, Room 221.

(6) Two spare room switches

7. The Registration File

There is one registration record of fifty-five bytes built for each student registered in each course segment. During the conversion process one registration record was created for each "5" card in the original punched-card system.

a. First group--identifier items

(1) Record delete flag

(2) Registration record key (sixteen characters of embedded key). Example 701MA32320201859

Where: 701 = Quarter number
MA3232 = Course number
02 = Segment number
01850 = Student record number

(3) Duplicate course flag. If "on" it means that this is the second time this course has been taken for

credit by this student. This flag is used to compute quality point average.

(4) Three spare switches

(5) Grade in course. This is three characters long and is initially set to blanks

(6) Input card number. This item is set when grade is recorded and can be used to retrieve the original grade card.

b. Second group--pointers. The record key itself is a master pointer for this record.

(1) Left pointer, student registration chain (used for student registration lists)

(2) Right pointer, student registration chain

(3) Left pointer, course registration chain (used for class rosters)

(4) Right pointer, course registration chain

The pointers for the above items have redundant information removed. The actual pointers are created by using the record key in conjunction with the information stored in group 2 items.

8. Thesis Title File

This file serves the dual role of being the source of the thesis titles as they are printed on academic records and also as a source for future information retrieval. The Registrar currently produces a list of theses each quarter, listing student name, thesis title, and advisor name. While the length of the record (301 bytes) is determined primarily

by the length of the longest possible thesis title (240 characters), this record can be considerably shortened (on average) when variable-length records are permitted by the next compiler version to be implemented at the computer center.

a. First group--identifier items

(1) Delete flag

(2) Record number (embedded key)

(3) Advisor (professor record number)

(4) Alternate advisor (professor record number)

(5) Quarter thesis approved

(6) Number of students submitting thesis

(7) Three spare switches

b. Second group--student items. These items are repeated fifteen times, large enough for massive group projects.

(1) Student record number

9. Entrance Credit File

All officer students have earned some academic credits elsewhere before attending the Naval Postgraduate School. When the student transcripts from prior institutions are received, the academic credits are evaluated and converted to "2" cards for the original punched-card system. These take the form of prior degrees awarded (for graduate students) and of credit hours (for baccalaureate students). During the conversion process an entrance credit record will be written for each "2" card in the original system. For

future updating purposes the portions of the "2" card applicable to entrance credits will continue to be used as the input source card.

a. File items--one group

(1) Delete flag

(2) Record key (embedded key). Example:

XXXXXYYZ where: XXXXX = Student record number
YY = Year (e.g., 65)
Z = Sub record number used for multiple credits in one year

(3) School code (four digits)

(4) Previous degree (or blank)

(5) Quarter hours of credit allowed (used if previous degree is blank)

(6) Two spare switches

10. Table File

The table file is used to store extended conversion tables and semi-permanent paragraphs for output forms.

a. File items--one group

(1) Delete flag

(2) Record key (embedded key--Form: AAXXXX)

(3) Table input data (YRMODA that record was inserted in file)

(4) Table input data (stored in character form--fixed format--65 characters)

B. THE ADVANTAGES AND DISADVANTAGES OF LIST PROCESSING FOR THIS APPLICATION

The advantages of using list processing were as follows:

1. Because the data remains relatively stable over a long period of time, and the relationships between data items are also stable, the list pointers provided a convenient method of avoiding repeated sorts of the same data over and over again. Without the registration chain pointers, for instance, the complete registration file would have had to be sorted every time class rosters were produced.

2. The pointers became a convenient link between files. Once a record is written on a disk, it stays in the same position. Only the pointers have to be changed as the relationships between data items change.

3. Each data item; such as names, titles, course hours, etc., appeared only once in the data base. Therefore, when these data items are changed, they automatically update all the reports and relationships that use the items. Without pointers the data items must be duplicated in many files, and the changing of any one item becomes more complicated.

4. Full advantage was made of the direct lookup feature of indexed sequential data sets.

The disadvantages of using list processing were as follows:

1. The programming of pointer update routines is more complicated than strict sequential record processing.

2. The insertion of records takes longer processing time because of the long walks through the chains to locate

the position of new records. This is counterbalanced by the avoidance of sorting time if the data is relatively stable.

3. Deletions of records are relatively simple and taken little processing time if both left and right pointers are maintained, but this increases the storage space allotted to pointers.

4. The software efficiency of the OS/360 indexed sequential-access method is not very great. Actual tests on extended updating runs using many files have shown that the core resident time can be kept within reasonable bounds provided the following measures were taken:

a. The highest level disk index was placed in core. This was accomplished by placing the word INDEXAREA in the environment attribute of each indexed file declaration. The core space for these files was small; for example, 765 bytes is the size of the highest level index for the student file.[13]

b. The indexed sequential files were dispersed to separate disk units to avoid seek-arm contention. For feasibility test purposes the dispersion was done from the "son" resident disk to other 2311 disk units. For production runs the dispersion should be to 2314 units to take

---

[13]This statistic, as well as other useful ones, were obtained from the "format" form of the volume table of contents (VTOC) listing provided by the IEHLIST utility program which was appended to the end of the feasibility program runs.

advantage of the shorter mean access times on that unit, provided that the data is stored on contiguous cylinders in the same unit. The higher data transfer rate of the 2314 should also speed up processing.

c. Master indexes were built at the time the larger files were created. This was accomplished by placing an M in the OPTCD sub-parameter of the DCB parameter of the files' job control language.

C. NAME-COMPRESSION HASH CODING USED FOR STUDENT RECORD NUMBER INDEX

The name-compression hash coding scheme was adapted from the scheme outlined in an article from the Communications of the ACM by Leon Davidson in March 1962, entitled "Retrieval of Misspelled Names in an Airline Passenger Record System."[14] Although the article describes the use of a name-compression system in searching for phonetic matches of names in an on-line computer system, the article also states that the name-compression system "lends itself to a nonphonetic ILL-SPELLED name search which is called upon whenever spelling errors more significant than vowel error, etc., are made."[15]

1. Name-Compression Algorithm Used

In order to compress student names, a procedure called COMPRESS was written which accepts any twenty-two-

---

[14]Davidson, Leon, "Retrieval of Misspelled Names in an Airline Passenger Record System," CACM, Vol. 5, pp. 169-171, March 1962.

[15]Ibid., p. 170.

character string and returns a five-character string. The algorithm follows these steps:

a. Initially set the resulting string to all banks

b. Separate the name string into a last name string and a first name or initial string by locating the first comma, if it exists

c. Place the first character of the first name (or initial) in the fifth character position of the resulting string. If no first name or initial exists, set the fifth character to "X."

d. The rest of the processing is performed only on the last name string. Begin by placing the first letter of the last name in character position 1.

e. Next take out all appearances of vowels, blanks, stray punctuations (,.-'), in the last name string and any occurrences of the letters H, W, and Y.

f. The remaining "squeezed" letters are checked for duplicate consecutive letters which are eliminated.

g. The 2nd, 3rd, and 4th characters are transferred to the result string.

2. Use of the Compressed Names in Generic Lookup

The same COMPRESS routine that was used to create the first five letters of the six-letter key of the student index file is used in the procedure LOOKUP-NAME which is entered with a twenty-two-character student name and sets the item ST-NAME-KEY to the student record number if it is found. If the record number is not found, the procedure

sets ST-NAME-KEY to blanks. Another global item, ST-NAME-OK, also indicates the result of the lookup search.

The search technique utilizes the generic key feature of PL/1 which applies only to index sequential files. If GENKEY is specified in the environment attribute of the file declaration, then any read statement with a key will return the first record in the file matching the number of characters in the key parameter. If an exact match (character by character) is not found, then the next higher key to the given one is returned. For the LOOKUP-NAME routine the initial generic key was set at the five characters of the compressed name. The full twenty-two name characters in the index record are compared to the entering name argument. If a match occurs, then the student has been found, and his record number is placed in ST-NAME-KEY and the procedure terminates. If no match is found, successive records are read sequentially[16] until the first five characters of the key do not agree with the compressed-name key of the entering argument. At this point the generic key argument is reduced to four characters, and the search is continued. Successive widening of the search domain is accomplished by decreasing the character width of the generic key. While the search is being performed, a "near-miss" table is built up which is used, in the case no record number is found, to print out error analysis lists. The search

---

[16]Sequential reads rarely require additional disk seeks because the student index records are blocked 112 records per block.

procedures outlined above are terminated when any one of the following occurs:

a. An exact match of name (twenty-two characters) is found

b. The near-miss table is filled with ten entries

c. The search has been widened to include only one generic key character.

3. Example of Results Obtained Using the Compressed-Name Algorithm and the Generic Ley Lookup Feature

Compressed names for all students placed on the student file were used to form the student index record. The unique sixth key character was formed by starting with 1, 2, 3, etc., followed by the sequence A, B, C, etc. In no case were the letters used, although this remains a possibility for the future. In the process of forming the registration and course records, the procedure LOOKUP-NAME was used to find the student record number. The following is an example of a printout where the search was unsuccessful. The entering argument name was "SCHAUMBURG, H. W." The resulting near-miss list was the following:

	Name	Record Number	Compressed Name Key
1.	SCHAUMBURG, HENRY W.	01740	SCMBH1
2.	SCHMIDT, CLIFFORD B.	01752	SCMDC1
3.	SCHIMMELS, JOHN N.	01747	SCMLJ1
4.	SCHUMANN, JAMES F.	01765	SCMNJ1
5.	SCHMITT, MICHAIL	01753	SCMTM1
6.	SCHEWE, NORMAN L.	01745	SC N1
7.	SCHADE, ERIC H. JR.	01735	SCD E1
8.	SCHEKDIG, ROBERT E.	01742	SCDGR1
9.	SECADES, VINCENT C.	01771	SCDSV1
10.	SCHUFELDT, CORAL V.	01761	SCFLC1

A similar result for the name "MCKENZIE, JOHN H. JR" was:

	Name	Record Number	Compressed Name Key
1.	MACKIN, JERE G.	01195	MCKNJ1
2.	MCKENDRICK, JOHN D. JR	01299	MCKNJ2
3.	MCKENZIE, JOHN H.	01300	MCKNJ3
4.	MCKINNEY, JAMES B.	01301	MCKNJ4
5.	MCKINNEY, JOHN W.	01302	MCKNJ5
6.	MACKENZIE, DONALD K.	01194	MCKND1
7.	MC KAY, DENNIS A.	01261	MCK D1
8.	MC KEE, DAVID L.	01262	MCK D2
9.	MCKAY, JOHN N., JR.	01295	MCK J1
10.	MCKEE, LESLIE L. III	01297	MCK L1

D. INITIAL LOADING PROCEDURE USED FOR DATA BANK

The following is an outline of the procedures used to set up the files initially. One of the peculiarities of the indexed sequential data set files is that the files can be opened for output writing only once. At this time the data is written in the prime data area, and the indexes for the prime data areas are initialized. Subsequently, the files must be opened for input (to the main program) or for update, in which case records can be added to the existing records. If no records reside in the prime data area and the file is opened for update, almost all of the records will be written in the overflow areas. Once the files are established, the data sets respond as expected, inserting records by forcing a few records into overflow areas. It is only the initial loading sequence that can be troublesome because of the large number of records initially written. For the large files the approach taken by this study was to

arrange the records initially in key order so that a large number of records could be written during the initial loading of the prime data areas.

1. Initial Loading of the Student Files

a. Master "J" cards from the original punched-card system were sorted on last name and the resulting data set used to write the student records and set the schoolwide alphabetical chain pointers.

b. Three data sets were created at the time the record number was assigned in (a), above, and two of the data sets were sorted and used to produce the pointers for curricular officer/alpha and curriculum/section/alpha links.

c. One data set produced in (a) was used to match against the punched-card file of the military personnel office to pick up SMC on most students, and set the SMC linkages.

d. One data set was used to create the compressed-name key. This augmented data set was sorted by key and used to create the student index file.

2. Initial Loading of the Professor File

Professor names from the three course registration files (original system) were extracted, sorted, and duplicates eliminated. Initials were added to the names, along with department codes, and the resulting cards sorted three ways to produce the basis for the professor records and the professor index file.

3. Initial Loading of the Course and Registration Files

a. Data sets on magnetic tapes were made of the course registration files at the end of quarters 1, 2, and 3 of Academic Year 1970-1971, and these tapes were used to set up the course and registration files. As a matter of interest for future production runs, this setup run was the most complicated attempted so far. Accessing four indexed sequential files simultaneously, writing one temporary data set on disk, reading three tape units (sequentially), producing error listings and error cards; the whole operation for 22,000 records was completed in seventy minutes with seven minutes of central processing time utilized.

b. Additional runs have been programmed to place the grades in the registration records from another two tapes of the academic record file from the original system. The entrance credit file will be written at the same time, picking up the original "2" card information in the academic record files.

4. Initial Loading of the Table File

The academic calendar table consisting of Academic Quarter start and stop dates was created from the school catalog, the school codes were converted from a punched-card file in the original system, and the paragraph tables for some of the output forms were all placed on punched cards in key number order and used to write the initial table file.

5. Initial Loading of Other Files

a. The initial loading of the master record file was performed by forming a structure in core identical to

the master record structure, initializing it to proper starting values for the master keys, and then writing the whole structure out with one write command.

b. The initial loading of the thesis title file will be delayed until the whole system is smoothly running.

VII. CONCLUSIONS AND FUTURE USES OF THE REGISTRAR'S ACADEMIC RECORD SYSTEM

This study has analyzed the punched-card system of the Registrar's Office at the Naval Postgraduate School and concluded that a more extensive utilization of the computer facilities available at the school could have these benefits:

1. Reduce processing time for academic records
2. Improve feedback of changes to academic records
3. Improve the data-retrieval capability of the academic record data
4. Improve the formats of the present reports and institute additional needed outputs[17]
5. Improve the disaster-recovery capability by storing the master disk files in two separate locations and by instituting a father-son-grandfather rotation between successive generations of disk files.

---

[17]Two features of the new system have already reached the production stage. The textbook checklist has been produced for the past two quarters. The first quarter it was produced directly from the "J" cards of the original system, and for the second quarter it was produced from both the "J" cards and disk student file. The other output is the grade cards which were run in parallel with the grade rosters for quarter 702 and as the sole source of grade reporting at the end of quarter 703. Grades were manually keypunched, however, from the new grade cards to the "5" cards of the original system instead of directly into the registration file of the new system. If enough effort is expended, it should be possible to read grades directly into the registration file by the end of quarter 704.

6. Institute an historical record system with built-in query capability

7. Rationalize the original system's input function and ease the burden of input error recovery

8. Ease the task of future system interfaces and file expansion

9. Provide the basis for a planning data bank in the academic record area.

Extensive programming and file setup have shown that the Registrar's information system can be operated within reasonable limits. Additional programming and systems effort are required to bring the system to full operation. As much as six man months of effort may be required to fully implement this system.

APPENDIX A

REGISTRAR'S GUIDE TO THE USE OF THE ACADEMIC RECORD COMPUTER SYSTEM

Preface

The outline for this guide was taken from Dorothy A. Walsh, A Guide for Software Documentation.[18] This guide should be considered a preliminary edition and should be extensively revised once the system is fully implemented.

Effective Use of This Guide

The computer system for the Registrar's data bank includes ten disk files and computer programs stored as a unit on one 2311 disk pack. Successive generations of computer runs produce additional disks which are updated versions of each other. Successive generations of one family of files are called GRANDFATHER, FATHER, and SON disks. For any one processing run the FATHER disk holds the copy of the files which is the file input source. This disk is immediately copied onto the SON disk before any updating commences. All updating is performed on or from the SON disk so that all data on the FATHER disk remains intact and can be used for making computer reruns. If both the FATHER and SON disks are lost or inadvertently erased,

---

[18] Walsh, Dorothy A., A Guide for Software Documentation, Advanced Computer Techniques Corporation, 437 Madison Avenue, New York, N.Y. 10022, 1969, p. 24.

the GRANDFATHER disk is held in reserve. The GRANDFATHER disk must also be retained until the FATHER disk has successfully created the next SON disk.

The purpose of this guide is to show how the computer system can be used to update the data files, query the data files, purge the data files, and produce outputs from the data files. The content of this guide includes

1. How input data is supplied to the computer programs.
2. What the capabilities of the system are.
3. What output is produced as a result of processing by the system.
4. How the output should be interpreted and used.

TABLE OF CONTENTS

SECTION B: HOW TO USE THE ACADEMIC RECORD COMPUTER SYSTEM

I. SPECIFYING THE INPUT DATA

All input data is submitted in the form of punched cards. There are two kinds of input cards: header cards and detail cards. Header cards are distinguished from detail cards because they have one of the following master KEYWORDS starting in column 1 of the card:

Master Keyword	Type of Function Performed
UPDATE	Activates file update routines
REPORT	Sets report calls to produce outputs
PURGE	Activates the file purge routines
QUERY	Activates the file query system

The remainder of the header card contains parameters depending on the type of function needed. All parameters on header cards are enclosed in parentheses groups: ( ). These parenthesis groups are not column dependent.

Detail cards follow header cards and can be submitted in large or small groups. As few as one or no detail cards is also permitted immediately following header cards. There is no requirement for sorting detail cards or for submitting the header cards in any specific order. Header and detail cards should be made up and assembled in the order transactions occur requiring computer functions. The computer processing works in such a manner that all file updates in a run batch are processed before any reports are produced or queries answered.

II. SPECIFYING THE PROCESSING DESIRED

In any of the specific processing requests described below two "optional parameters" may be added. These are the QTR parameter and the LIMIT parameter.

The QTR parameter is specified by (QTR = XXX) where XXX is a 3-digit number of the form of academic year (2 digits) and quarter (1 digit). For example 703 would specify the third quarter of academic year 1970-1971. 711 would specify the first quarter of academic year 1971-1972 which begins in July, 1971 and ends in September. If the quarter parameter is not given the computer program assumes that the applicable quarter is the present quarter. For computer processing purposes the inter-quarter break is assumed to belong to the previous quarter.

The LIMIT parameter is specified by (LIMIT = XXXXX) where XXXXX is any integer of length 1 to 5 digits. The purpose of this parameter is to set an upper limit (usually for test purposes) on the number of outputs desired from a particular report or other function. If the LIMIT parameter is not given the computer program default of 30,000 is applied as the upper limit.

The header cards for the UPDATE functions are the most complicated and will be discussed first.

UPDATE header cards are constructed by referring to one of the following tables which appear at the end of this guide:

1. Student File Update Functions
2. Professor File Update Functions
3. Course File Update Functions
4. Registration File Update Functions
5. Thesis File Update Functions
6. Entrance Credit Update Functions
7. Table File Update Functions
8. Master File Update Functions

A few examples will be given to demonstrate the use of these tables.

Example 1:

Suppose the projected rotation date on LCDR John P. Smith needs to be changed to September, 1973. Referring to the first update function table the following header and detail cards would be correct:

Header card: UPDATE (CHANGE: PRD)

Detail card: (SMITH, JOHN B.) (0973)

In this case the detail card could have been in either of 3 formats, the type "J" card, type 3 or type 4. Type 3 detail card was demonstrated above. If it is more convenient to use a copy of the "J" card showing the officer's name and new PRD then this type may be used. Any number of detail cards may follow the header cards and the type of detail card formats may be intermixed. If the "J" card is used (as for the example above) only those columns applicable to the function asked for will be scanned by the computer program. The other columns of the card need not be filled out.

Example 2:

Suppose a new professor, A. B. GAMMA, is assigned to the mathematics department (organization code 53) and is to begin teaching classes next quarter. His mail delivery code is 53ZQ and his academic rank is Associate Professor (Code = 5). Referring to the third table at the end of the guide the correct input cards would be:

```
Header card:  UPDATE (ADD: P-REC)
Detail card:     (GAMMA, A. B.)  (53ZQ) (5)
```

The number of leading and trailing blanks within the parenthesis group is not critical nor is the placement of the parenthesis group on the cards; any card column will do. If illegal combinations of functions or the incorrect data type is used, the input checking routine will return the incorrect card in the form of an exact duplicate of the card that was not processed along with an "OOPS" listing entry that shows what the problem is.

STUDENT FILE UPDATE FUNCTIONS

Keyword	Item	# Char	Add	Change	Delete	Detail Card Formats
S_REC	Whole student record	--	YES*	NO	YES	"J", 1,2
S_NAME	Student name	22	NO	YES	NO	"J", 3,4
RANK	Officer rank (abbrev)	6	YES	YES	NO	3,4
SECT	Student section	4	YES	YES	YES	"J", 3,4
CORP	Officer Corp abbrev	5	YES	YES	YES	3,4
SMC	Student Mail Center Box #	4	YES	YES	YES	"J", 3,4
STAFFCODE	Mail delivery if no SMC	4	YES	YES	YES	3,4
OBLISERV	Obligated service start date (MOYR) (input quarter start date)	4	YES	YES	YES	"J", 3,4
PRD	Prospective rotation date (est. grad date) (MOYR)	4	YES	YES	YES	"J", 3,4
DESIG	Officer designator	4	YES	YES	YES	"J", 3,4
SERIAL	Officer serial number	6	YES	YES	YES	"J", 3,4
SSN	Social Security number	9	YES	YES	YES	"J" 3,4
COUNTRY	Country code	2	YES	YES	YES	3,4
EDCODE	Education code/change of curriculum	4	YES	YES	YES	5,6
D_TYPE	Degree to be awarded (type)	2	YES	YES	YES	3,4
D_QTR	Qtr. that degree awarded	3	YES	YES	YES	3,4
B_FWD	Credits brought forward	-	YES	YES	YES	"4"

*Use J Card format only.
Card Formats are on following page (97).

STUDENT FILE UPDATE FUNCTIONS (Continued)

CARD FORMATS:

"J" card has a J in column 1 and follows format of punched card system.
"4" card has a 4 in column 1 and follows format of punched card system.
Type 1 format is: (LASTNAME, FIRST X.)
Type 2 format is: (S = XXXXX) where XXXX is student record #
Type 3 format is: (LASTNAME, FIRST X.) (New data)
Type 4 format is: (S = XXXXX) (New data)
Type 5 format is: (LASTNAME, FIRST X.) (NEW DATA) (QTR = XXX)
Type 6 format is: (S = XXXXX) (New data) (QTR = XXX)

PROFESSOR FILE UPDATE FUNCTIONS

Keyword	Item	# Char	Add	Change	Delete	Detail Card Formats
P_REC	Whole professor record	--	YES*	NO	YES	7,8,9
P_NAME	Professor name	23	NO	YES	NO	8,9
P_DEPT	Professor department	3	YES	YES	NO	8,9
P_CODE	Professor Original code	2	YES	YES	NO	8,9
P_RANK	Professor rank code	1	NO	YES	NO	8,9

*Use card format type 7 only.

Type 7 format is: (LASTNAME, X. Y. Z.)(54AB) (3) where 54AB is dept & code and 3 is academic rank code.
Type 8 format is: (LASTNAME, X. Y. Z.)(New data).
Type 9 format is: (P = XXXX) (New data) where XXXX is the professor record number.

COURSE FILE UPDATE FUNCTIONS

Keyword	Item	# Char	Add	Change	Delete	Detail Card Formats
C_REC	Whole course record	--	YES*	NO	YES	"A", 12
TITLE	Course title	22	NO	YES	NO	10
C_PROF	Course primary professor	--	NO	YES	NO	11
C_ALTPROF	Course alternate professor	--	YES	YES	YES	11
C_R_HRS	Course lecture hours	1	YES	YES	YES	10
C_L_HRS	Course lab hours	1	YES	YES	YES	10

*Type "A" card is a course header card under punched card system.

Type 10 format is: (MA3232) (New data) section 1 assumed
or (MA323202) (New data) section 2 specified.
Type 11 format is: (MA3232) (LASTNAME, X. Y. Z.)
or (MA3232) (P = XXXX).
Type 12 format is: (MA3232) or (MA323202).

REGISTRATION FILE UPDATE FUNCTIONS

Keyword	Item	# Char	Add	Change	Delete	Detail Card Formats
R_REC	Whole registration record	--	YES	NO	YES	10, "5"
MARK	Course mark (grade)	3	YES	YES	NO	"G"

Type "5" has a 5 in column 1 and follows format of punched card system

Type "G" is the new grade card format. The columns for 'student registration index number' are optional. If they are filled out these columns contain the student's record number.

Type 10 format is: (LASTNAME, FIRST X.) (Course #) If the course number includes the segment number (2 digits) the form is: MA323202; if the segment is not included segment 1 is automatically assigned: MA3232.

THESIS FILE UPDATE FUNCTIONS

Keyword	Item	# Char	Add	Change	Delete	Detail Card Formats
T_REC	Whole thesis record	--	YES	NO	YES	11
T_TITLE	Thesis title	240	NO	YES	NO	12
T_ADVISOR	Thesis Advisor	23	NO	YES	NO	12
T_ALTADVISOR	Thesis alternate advisor	23	YES	YES	YES	12
T_STUDENT	Thesis student	22	YES	NO	YES	12

Type 11 format is: (Thesis title) (P = XXXX) (P = XXXX) (S = XXXXX)

Notes: The thesis title may be continued on as many as 3 cards with the start of the title indicated with the left parenthesis and the end marked with the right parenthesis.* The name form of the professors' and student(s) entry may also be used. If the second professor is omitted then there is no alternate advisor. Up to 15 students may be included on one series of cards. The evidence of a "continuation card" is any character to the right of the rightmost parenthesis on the preceeding card.

Type 12 format is: (T = XXXXX) (New data) where XXXXX is an existing thesis record number.

Note: If the quarter approved is not the current quarter then the optional parameter (QTR = XXX) should be used.

*In order to insert a parenthesis as part of the thesis title text, use a repeated parentheses group:)).

BIBLIOGRAPHY

1. Brooks, F. P., and Iverson, K. E., Automatic Data Processing System/360 Edition, Wiley, 1969

2. Brown, G. D., System/360 Job Control Language, Wiley, 1970.

3. Davidson, L., "Retrieval of Misspelled Names in an Airlines Passenger Record System," Communications of the ACM, V. 5, p. 168-171, March 1962.

4. Humphrey, A. L., and Munro, W. G., "Management Information Retrieval," The Computer Journal, V. 13, p. 127-130, May 1970.

5. IBM Technical Manual C20-1649-2, Introduction to IBM System/360 Direct Access Storage Devices and Organization Methods, October 1967.

6. IBM Technical Manual C28-8201-1, IBM System/360 PL/1 Reference Manual, Second Edition, March 1968.

7. IBM Technical Manual C28-6594-4, IBM System/360 Operating System PL/I (F) Programmer's Guide, Fifth Edition, November 1968.

8. Laden, H. N., and Gildersleeve, T. R., System Design for Computer Applications, Wiley, 1965.

9. Martino, R. L., MIS-Methodology, Management Development Institute, Wayne, Pennsylvania, 1969

10. Naval Postgraduate School Catalogue for 1970-1972.

11. Naval Postgraduate School Integrated Operating and Development Plan 70, November 1969.

12. Walsh, D. A., A Guide for Software Documentation, Advanced Computer Techniques Corp., 1969.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Dean B. J. Lockhart Dean of Curricula, Code 022 Naval Postgraduate School Monterey, California 93940	1
4.	Registrar, Code 0221 Naval Postgraduate School Monterey, California 93940	3
5.	LCDR E. A. Singer, USN, Code 53SF Department of Mathematics Naval Postgraduate School Monterey, California 93940	1
6.	LCDR C. D. Vincent, USN USS PERMIT (SSN-594) S.F. Bay Naval Shipyard Vallejo, California 94590	1
7.	CDR J. D. Hartley, USN Operations Research Curricular Officer, Code 30 Naval Postgraduate School Monterey, California 93940	1
8.	Director, W. R. Church Computer Center, Code 0211 Naval Postgraduate School Monterey, California 93940	4
9.	Administrative ADP Coordinator, Code 0329 Naval Postgraduate School Monterey, California 93940	1
10.	LCDR W. L. Powell, USN USS GUITARRO (SSN-665) S.F. Bay Naval Shipyard Vallejo, California 94590	1

11. LCDR W. E. Rickman, USN 1
USS KAMEHAMEHA (SSBN-642)
FPO New York, New York 09501

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author)

Naval Postgraduate School
Monterey, California 93940

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3 REPORT TITLE

THE DESIGN AND IMPLEMENTATION OF THE REGISTRAR'S INFORMATION SYSTEM AT THE NAVAL POSTGRADUATE SCHOOL

4 DESCRIPTIVE NOTES (Type of report and, inclusive dates)

Master's Thesis; March 1971

5. AUTHOR(S) (First name, middle initial, last name)

Charles David Vincent, Lieutenant Commander, United States Navy

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March 1971	105	12

8a. CONTRACT OR GRANT NO.

b. PROJECT NO.

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10 DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

Naval Postgraduate School
Monterey, California 93940

13. ABSTRACT

The design and implementation of a new information system for the Registrar's office at the Naval Postgraduate School was preceded by an analysis of the original punched-card record-keeping operation. Major goals for the new system included rapid grade reporting, feedback reports to increase data reliability, file security and disaster recoverability, data retrieval from current and historical data bases, and ease of input-output batch processing. Implementation was accomplished on an IBM System/360 Model 67 operating under OS/360. The information system utilized direct access storage devices and list-processing methods. Additional reports for professors, administrators, and students are planned utilizing the course, registration, professor, student, entrance-credit, and thesis-title files maintained on direct-access disk units. File expansion and future interfaces for the Registrar's information system are discussed.

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
INFORMATION SYSTEM						
REGISTRAR						
GRADE REPORTING						
HISTORICAL DATA RETRIEVAL						
IBM SYSTEM/360						
DASD (DIRECT ACCESS STORAGE DEVICES)						
LIST PROCESSING						
REGISTRATION RECORDS